# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 09/545,493 | 04/07/2000 | Rogier Pierre | T2147-906388 | 9721 |

181 7590 04/13/2005

MILES & STOCKBRIDGE PC
1751 PINNACLE DRIVE
SUITE 500
MCLEAN, VA 22102-3833

| EXAMINER | |
|---|---|
| SHAH, NILESH R | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2195 | |

DATE MAILED: 04/13/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

| Office Action Summary | Application No. | Applicant(s) | |
|---|---|---|---|
| | 09/545,493 | PIERRE, ROGIER | |
| | Examiner | Art Unit | |
| | Nilesh Shah | 2195 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☐ Responsive to communication(s) filed on *02 February 2005*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *24-49* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *24-49* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

## DETAILED ACTION

1. Claims 24-49 are presented for examination.

### *Claim Rejections - 35 USC § 103*

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> a. A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 24-49 are rejected under 35 U.S.C. 103(a) as being unpatentable over Ogi (5,854,938) in view of Kimmel et al (6,105,053) (hereinafter Kimmel).

4. As per claim 24, Ogi teaches the invention substantially as claimed including a process for assigning tasks to a processor in a multiprocessor digital data processing (col. 1, lines 30-34) system having preemptive operating system, and a given number of processors capable of processing said tasks in parallel (col. 1, lines 30-34), comprising in at least one preliminary phase dividing said processors into groups each group comprising predetermined numbers of processors (col. 3, lines 47-55).

5. Ogi does not specifically teach the use of a predetermined number of queues. Kimmel teaches dividing said tasks into a predetermined number of elementary task queues and storing a predetermined number of tasks to be processed in a given priority (col. 19 lines 44-46;col. 2 lines 47-57;col. 10 lines 63-67col. 14 lines 44-65) in each elementary task queue each of said processor groups being associated with an elementary task queue (col. 12, lines 17-24; col. 11, lines 23-30) each of the stored predetermined number of tasks being associated with one of the processors associated with said elementary queue (col. 2, lines 47-55; col. 19, lines 43-55).
6. It would have been obvious to one skilled in the art at the time of the invention to combine the teachings of Kimmel and Ogi because Kimmel teaching of having each queue in a predetermined manner it would improve Ogi's system by making the selection of next jobs more efficient and easier to understand.

7. As per claim 25 Ogi teaches a wherein said processor groups each comprise an identical number of processors (col. 9 lines, 40 –49).

8. As per claim 26 Kimmel teaches a process comprising generating a series of tests and measurements in an additional preliminary phase for determining the number of processors in each group and the number of groups for achieving the best performance of said system (col. 10, lines 34-49; col. 16, lines 35-45).

9. As per claim 27, Ogi teaches a system is constituted by a predetermined number of modules linked to one another, each comprising a given number of processors and storage means, each of said processor modules constituting one of said groups (col. 8, lines 57-67; col. 3, lines 47-55).

Kimmel teaches a process wherein the architecture of said system is of the non-uniform memory access type (col. 4, lines 18-25) and
each module being associated with one of said elementary task queues of an associated processor (col. 2, lines 47-55; col. 19, lines 43-55).

10. As per claim 28, Kimmel teaches a process further comprising associating each of said processors with a first data structure for identification of the associated processor, said first data structure comprises at least one first set of pointers (col. 31, lines 39-44), associating said first set of pointers with one of said elementary task queues, associating each of said elementary task queues with a second data structure, said second data structure having at least one second set of pointers (col. 19, lines 10-20; col. 31, lines 39-44), associating said second data structure with one of said processor groups, storing all of the tasks to be processed in said system in a table, each of said second data structures of the elementary queues further comprising a third set of pointers, said third set of pointers each associating elementary task queues with one of said tasks stored in the table(col. 19, lines 10-20) or with a series of concatenated tasks, and associating each of said tasks of the table with a third data structure that comprises a fourth set of pointers

said fourth set of pointers associating third data structure with one of said elementary queues (col. 19, lines 10-20; col. 31, lines 39-44).

11. As per claim 29, Kimmel teaches a process comprising distributing said tasks among said elementary queues in at least one additional phase by searching, when a new task to be processed is created, for a queue with the lightest load among all of said elementary queues of said system and assigning said new task to said elementary queue with the lightest load so as to balance the global load of said system among said elementary queues (col. 15, lines 45-55; col. 16, lines 40-42).

12. As per claim 30, Kimmel teaches process further comprising performing said distribution of tasks by determining a composite load parameter associated with each of said elementary queues associating each processor with a memory, calculating said composite load parameter as the sum of the load of a processor or a processor group associated with said elementary queue and the load of the memory associated with said processor or processor group (col. 19, lines 10-20; col. 31, lines 39-44).

13. As per claim 31, Kimmel teaches a process comprising checking in a preliminary step whether said task is linked to one of said elementary queues, and when said test is positive, assigning said linked task to the elementary queue group (col. 10, lines 34-49; col. 16, lines 35-45).

14. As per claim 32, Kimmel teaches a process comprising at least one additional phase and searching for a remote elementary queue that is not empty when one of said elementary queues associated with one of said processor groups is empty of executable tasks selecting in said empty elementary queue a task executable by one of said processors of said processor group associated with the empty elementary queue and transmitting said selected task to said one of said processor for processing so as to globally balance the processing of said tasks in said system (col. 12, lines 7-14; col. 16, lines 35-55).

15. As per claim 33 Kimmel teaches a wherein said non-empty elementary task queue has a predetermined minimal occupation threshold (col. 16 lines 46-57, col. 10 lines 35-47).

16. As per claim 34, Kimmel teaches the use of a process further comprising storing the tasks in decreasing order of priority, skipping a predetermined number of tasks before scanning the other tasks of said non-empty elementary queue in order to search for an executable task and have said executable task processed by one of said processors of said processor group associated with the empty elementary queue (col. 9, lines 9-17; col. 10, lines 8-13).

17. As per claim 35, Kimmel teaches a process wherein said number of skipped tasks and the maximum number of scanned tasks among all tasks stored in said non-empty elementary queue are variable over time and are determined by a self-adapting process from the number of tasks that are or are not found during said scans and from the position of these

tasks, sequenced in order of priority, in said non-empty elementary queue (col. 11, lines 41-50; col. 11, lines 63-65).

18. As per claim 36, Kimmel teaches a process wherein said selected task is associated with a minimal value of a cost parameter (col. 16, lines 50-57), which measures global performance degradation of said system due to the processing of said selected task in said non-empty remote elementary queue (col. 12, lines 7-14) by one of said processors of said processor group associated with the empty elementary queue (col. 11, lines 42-50).

19. As per claim 37, Kimmel teaches a process comprising periodically measuring for a balanced distribution of said tasks in said elementary queues in at least one additional phase and when an unbalanced state of said system is determined, selectively moving tasks from at least one elementary queue with a heavier load to an elementary queue with a lighter load (col. 16, lines 36-55).

20. As per claim 38, Kimmel teaches a process further comprising discontinuing the step of selectively moving tasks when said imbalance is below a certain threshold (col. 16, lines 46-57; col. 10, lines 35-47).

21. As per claim 39, Kimmel teaches a process wherein all or some of said tasks belong to multitask processes, and each multitask process requires a given memory size and workload, further comprising measuring workloads and memory sizes, in the system and

selecting the process requiring the greatest workload and the smallest memory size, and moving all the tasks of said selected process to the elementary queue with the lightest load (col. 16, lines 36-55).

22. As per claim 40, Kimmel teaches a process further comprising it comprises a preliminary step of checking whether all tasks of said multitask process that must be moved belong to the elementary queue set with the heaviest load and whether any task is linked to any of said processor groups (col. 16, lines 36-55).

23. As per claim 41, Ogi teaches a process further comprising said preemptive operating system is used in a server in a distributed network environment (col. 1, lines 25-27).

24. Claim 42 is rejected based on the same rejection as claim 24 above.

25. As per claim 43, Kimmel teaches means for determining the load of said elementary queues and for assigning a new task created in said system to the elementary queue with the lightest load (col. 16, lines 36-55).

26. As per claim 44, Kimmel teaches an architecture further comprising, when one of said elementary queues associated with one of said processors is empty, means for locating a non-empty, remote elementary queue, and an executable task in said non empty elementary queue, and assigning said executable task to said one of said processor for

processing said executable task (col. 12, lines 7-10; col. 11, lines 40-47; col. 24, lines 50-55).

27. As per claim 45, Kimmel the use of an architecture further comprising means for detecting an imbalance between elementary queues, and for determining when an imbalance is detected the elementary queue with the heaviest load and the elementary queue with the lightest load, and means for moving tasks from the elementary queue with the heaviest load to the elementary queue with the lightest load (col. 16, lines 36-55).

28. As per claim 46, Kimmel teaches an architecture wherein the operating system of the processing system is of the nonuniform memory access type (col. 1, lines 5-8), and comprises modules linked to one another, each module comprising a given number of processors and storage means, each of said modules constituting one of said groups, each module being associated with one of said elementary queues (col. 19, lines 10-20; col. 9, lines 45-50).

29. As per claim 47, Kimmel teachers an architecture wherein the operating system of the processing system is of the nonuniform memory access type (col. 1, lines 5-8), and comprises modules linked to one another, each module comprising a given number of processors and storage means, each of said modules constituting one of said groups, each module being associated with one of said elementary queues (col. 19, lines 10-20; col. 9, lines 45-50).

30. As per claim 48, Kimmel teaches an architecture wherein the operating system of the processing system is of the nonuniform memory access type (col. 1, lines 5-8), and comprises modules linked to one another, each module comprising a given number of processors and storage means, each of said modules constituting one of said groups, each module being associated with one of said elementary queues (col. 19, lines 10-20; col. 9, lines 45-50).

31. As per claim 49, Kimmel teaches an architecture wherein the operating system of the processing system is of the nonuniform memory access type (col. 1, lines 5-8), and comprises modules linked to one another, each module comprising a given number of processors and storage means, each of said modules constituting one of said groups, each module being associated with one of said elementary queues (col. 19, lines 10-20; col. 9, lines 45-50).

***Response to Arguments***

32. Applicant's arguments filed 2/2/05 have been fully considered but they are not persuasive. Applicant states Kimmel does not teach the use of predetermined number of tasks. However Kimmel clearly teaches the use of predetermined policy considerations for allocation (col. 19 lines 44-46;col. 2 lines 47-57;col. 10 lines 63-67col. 14 lines 44-65).

***Conclusion***

33. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Nilesh Shah whose telephone number is (571)272-3771. The examiner can normally be reached on 9-5.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng An can be reached on (571)272-3756.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Nilesh Shah
Examiner
Art Unit 2127

NS
September 28, 2004

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100